A cognitive computational model of eye movements investigating visual strategies on textual material

Benoît Lemaire, Anne Guérin-Dugué, Thierry Baccino, Myriam Chanceaux, Léa Pasqualotti

A cognitive computational model of eye movements investigating visual strategies on textual material

Benoît Lemaire (Benoit.Lemaire@upmf-grenoble.fr)
LPNC, CNRS & University of Grenoble, France

Anne Guérin-Dugué (anne.guerin@gipsa-lab.grenoble-inp.fr)
Gipsa-lab, CNRS & University of Grenoble, France

Thierry Baccino (Thierry.Baccino@univ-paris8.fr)
Lutin Userlab, CNRS & Cité des Sciences et de l'Industrie, Paris, France

Myriam Chanceaux (Myriam.Chanceaux@univ-provence.fr)
LPC, CNRS & University of Provence, Marseille, France

Léa Pasqualotti (pasqualotti@lutin-userlab.fr)
Lutin Userlab, CNRS & Cité des Sciences et de l'Industrie, Paris, France

Abstract

This article presents a computational model of the visual strategies involved in processing textual material. An experiment is presented in which participants performed different tasks on a multi-paragraph page (searching a target word, searching the most relevant paragraph according to a goal, memorizing paragraphs). The proposed model predicts eye movements based on 5 parameters. The weighting of parameters is determined for each task by means of a multidimensional comparison of participant and artificial scanpaths.

Keywords: Computational model; Eye movements; Visual strategy; Text.

Introduction

Reading a text is a complex task which has been widely studied in cognitive science. Several models have been proposed to account for the peculiarities of human eye movements and especially the sequence of fixations and saccades that can be nowadays easily observed and recorded. For instance, EZ-Reader (Reichle, 2003) proposes a detailed model of how low-level processes such as oculomotor control, attention, visual processing and word identification combine to produce a relevant scanpath. In addition to a theoretical framework, EZ-Reader offers a computational model which can be run on a specific text.

Those models are models of reading. A typical reading scanpath is a sequence of short forward saccades followed by a long backward saccade going to the beginning of the next line, then short forward saccades, etc. until the end of the text. Not all words are fixated and there can be short regressive saccades (up to 20% of all fixations) but the general shape looks like that. However, texts can be processed in different ways: when you are searching information on a web page, not all the words of all the lines are processed. Sometimes, a specific word tells you that the current sentence is probably not relevant and you jump a few lines. You can also quickly choose to abandon the current paragraph and move to another one.

Another way to process a text is to search for a particular word. The scanpath then looks even more different: only some words are fixated in a very fast browsing of the text.

However, if you read to learn the text, you will show short forward saccades as usual, but also a high proportion of regressive saccades, even moving to previous lines, in order to make sure that information is correctly stored in memory. Simola et al. (2008) showed that different tasks on textual material produce different kind of scanpaths.

Carver (1990) distinguished five kinds of processes (visual strategies), based on variations of reading rates:

- *Scanning* is performed at 600 words/min and is used when readers are looking for a particular word;
- *Skimming* is used when readers need to get a quick overview of the content of the text (450 words/min.);
- *Rauding* is normal reading (300 words/min.);
- *Learning* is performed at 200 words/min. It is used when readers try to acquire knowledge from the text;
- *Memorizing* is used when readers want to memorize the text, therefore constantly verifying that information have been memorized (138 words/min.).

These processes differ in reading rates, but also in the length of saccades, fixation durations and number of regressions.

The aim of the present study was to design a cognitive computational model of eye movement that would account for all these strategies. The idea is to base this model on a very small number of parameters that can generate this variety of scanpaths, when appropriately tuned. The first purpose is to know the contribution of each of these variables in the production of the scanpath. For example, the spatial distance to the next fixation (saccade amplitude) is a key variable in rauding (words that are spatially close are much more likely to be selected than distant words) whereas it is not as important in scanning.

The second goal is to produce a general model of eye movements on texts which could easily adapt to high-level changes. For instance, a user may be looking for some information, first engaging in a skimming task, then switching to a learning process for a while, then moving to a scanning process because a specific word that occurred previously has to be reread in context. Our claim is that these processes are along a continuum. It is therefore interesting to model this behavior in a continuous way.

In order to build the model, we first gathered experimental data on different ways of processing a text.

Experiment

Procedure

An experiment in which participants would generate various kinds of scanpaths was designed. Three tasks were defined:

- Searching for a particular word in the page. This task is likely to generate scanning scanpaths.
- Searching among a set of paragraphs the one which best matches a given goal. For instance, if the goal is “planet observation”, the participant has to select the paragraph which is about that topic, although the paragraph may not contain those words: search has to be done based on semantics. In order to obtain rich scanpaths, several paragraphs may correspond to the goal; participants have to select the closest one. This task is likely to generate skimming scanpaths.
- Reading paragraphs in order to be able to answer comprehension questions afterwards. This task is likely to generate memorizing scanpaths.

Only 3 of the 5 processes defined by Carver were used, but, as we show later, the proposed model is not limited to them.

Materials

20 pages were generated in French. Each page was associated with a specific goal (for the skimming task). Examples of goals were *tribunal international* (international tribunal), *réhabilitation des logements* (housing renovation), *associations humanitaires* (humanitarian associations), etc. One target word per page was defined for the scanning task.

Seven paragraphs were produced for each page. In order to control the semantic relatedness of paragraphs to goals, Latent Semantic Analysis (Landauer et al., 2007) was used, a method to compute semantic similarities between texts.

LSA was trained on a 24 million word French corpus composed of all articles published in the newspaper “Le Monde” in 1999. A 300 dimension space was generated from the corpus, by means of a singular value decomposition of the word x paragraph occurrence matrix (see. Martin & Berry (2007) for more details). Each word of the corpus being represented as a 300 dimension vector, new texts can also be represented as vector by means of a simple sum of their words. A cosine function was used to compute

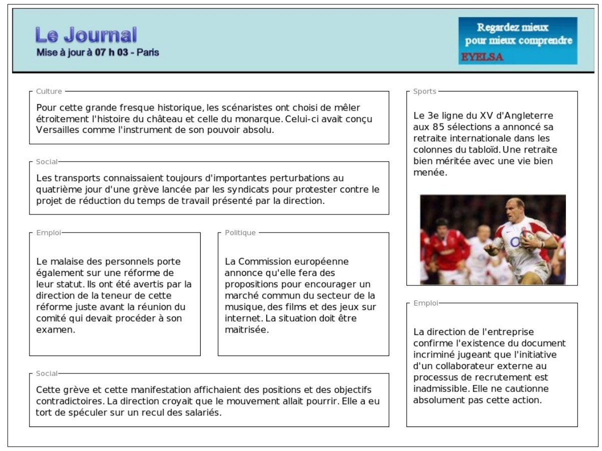

Figure 1: Example of page used in the experiments.

the similarity between vectors. The higher the cosine value, the more similar the two texts are.

From the seven paragraphs designed, two were highly related with the goal (cosine with the goal above .40), two were moderately related (cosine between .15 and .30) and three were unrelated (cosine below .10). In order to have a more realistic situation, an image was also included in the page as well as a banner. Figure 1 presents an example of a page. All paragraphs were organized into the page according to a layout that was randomly selected. There were eight versions of each page, in order to ensure that paragraphs are not processed in the same order.

Because the exact coordinates of words were needed for simulations, all pages were generated by a piece of software of our own which generates the image file and the word coordinates. The font was BitstreamVeraSans 12pt.

Participants

13 participants were recruited in the scanning condition, 8 in the memorizing condition, 34 in the skimming condition. All participants saw the 20 pages in random order. All scanpaths were recorded using a SR Research Eyelink 2 eyetracker. The images were presented on a 19 inch CRT monitor at a viewing distance of 50 cm.

Model

The main issue of the current model was to select which word to fixate next among all words in the paragraph, using a limited number of variables. That problem can be viewed as an iteration of two steps: weighting all words and selecting the best weighted one.

There are two ways for a variable to weight words: either by increasing the weight values of words likely to be fixated or by decreasing the weight values of words that will probably not be fixated. Some variables thus aim at selecting interesting words, other decrease the weight value of uninteresting words.

In order to present the variables used, let us describe how the different processes operate. Each process will correspond to a specific combination of these variables[1].

Scanning

Scanning is the fastest strategy. The aim being to find a particular word (the target), it is likely that users tend to prefer words which match with that target. Since almost all words can only be viewed in peripheral vision, the weighting can only be done on similarity of shape by a kind of pattern-matching process. **Shape similarity with the target** is therefore the first variable. This variable will probably not be used by the other processes which do not rely on a target word. In addition, it is likely that the scanning process shows longer saccades compared to rauding. The hypothesis is that the closer the process is to classical reading, the shorter the saccades. **Distance to the current fixation** is therefore our second variable: words spatially close to the current fixation will be preferred. Scanning is a process which will probably not need a high weight to that variable, as opposed to rauding for example.

Skimming

Skimming differs from scanning in that it takes the content into account. However, not all words need to be fixated in order to keep a high processing speed. For the same reason as before, the decision to select a word or not can only be done under peripheral vision. Although the general shape of a word is certainly not related to its meaning, it is likely that users tend to prefer long words which are known to be more meaningful. **Word length** is our third variable. It is possible that others processes rely on that variable, but probably to a lesser extent than the skimming process.

Rauding

Rauding is normal reading. Almost all words have to be fixated. Therefore, the linear sequence of words becomes important in order to preserve the meaning of sentences. Saccades towards the next word tend to be the rule. These saccades are therefore mostly horizontal (including the long saccade going to the beginning of the next line). **Saccade horizontality** is therefore our fourth variable: it would give higher weights to words reachable with an horizontal saccade. Scanning would probably give a low weight to that variable because saccades may jump from one line to another. Instead, the number of intervening words between the previously fixated word and the current fixated word could have been used as a variable. That value would be close to 0 in rauding, larger than 0 but positive in skimming and sometimes negative in memorizing. However, that variable would not have captured the fact that in 2D fixating a distant word in the text may result in a short saccade.

[1]It is important not to confuse variable weighting with word weighting. To sum up, a given process (scanning, rauding, etc.) assigns predefined weights to variables. Then each word is given a weight by simply combining the values given by all variables.

Learning

Learning falls in-between rauding and memorizing. This process is slower than rauding because of longer fixation durations, and more regressive saccades. As in skimming, word length should play a role. However, almost all the words should be fixated, saccade horizontality should also been involved.

Memorizing

Memorizing is the slowest way of reading. Almost all words have to be fixated but, as opposed to *rauding*, they might be fixated more than once (rauding may also involve regressive saccades on the previous word but we are here talking about long regressive saccades). On the other hand, there is almost no fixation on previously fixated words in the other processes. Therefore, the fifth variable is called **newness**. This variable prefers words that have not been fixated previously. The memorizing process is therefore likely to give a low weight to that variable in order to select words that were seen before. Other processes will probably give higher weight to that variable.

To sum up, the model assigns a weight to all words of the text and moves to the one with the highest weight. Given the current fixation C, the weight $w(W_i)$ of a word W_i depends on the following parameters.

shapeSim(W_i): the visual similarity between W_i and a target word, if any. This similarity between words should not be based on the identity of letters (not processed in peripheral visual field) but rather on the identity of shapes. Therefore, each word corresponds to a string in which each letter is represented by a character denoting its class (b=lowercase ascender letter, g=lowercase descender letter, a=lowercase normal letter, A=uppercase letter). For instance, the word Psychology is represented as Aagababagg. Similarity of shapes is performed by computing the Levenshtein (1966) distance between these strings. For instance, the distance between Psychology and Intrepidity (Aabaagababg) is 4 because four operations are needed to transform one string into the other (3 substitutions and 1 insertion). This distance is normalized for the longest string. Shape similarity is one minus that distance.
dist(W_i): the spatial distance between C and W_i, normalized for the length of the paragraph diagonal (longest saccade ever).
length(W_i): the number of characters of W_i, normalized for the longest word in the paragraph;
hor(W_i): the horizontality of $\overrightarrow{CW_i}$ defined as the angle between an horizontal line and the vector, normalized for $\pi/2$;
newness(W_i): a binary variable which is 0 in case W_i has already been fixated and 1 otherwise.

The general formula is:

$$w(W_i)=w_S.\text{shapeSim}(W_i) + w_D.\text{dist}(W_i) + w_L.\text{length}(W_i) + w_H.\text{hor}(W_i) + w_N.\text{newness}(W_i)$$

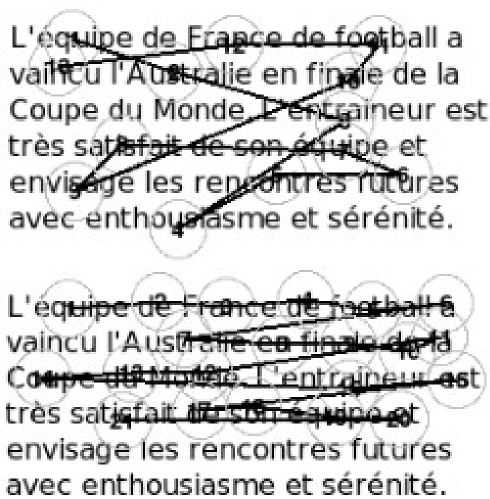

Figure 2: Two artificial scanpaths with parameters 10/10/10/100/10 and 0/10/0/200/100.

The model is not deterministic because in case two words are equally weighted, a random choice is performed.

Examples of scanpaths

Some examples of scanpaths using different weights are described in Figure 2. The model shows the upper scanpath with w_S=10, w_D=10, w_L=10, w_H=100, w_N=10. It looks like a scanning scanpath. However, with w_S=0, w_D=10, w_L=0, w_H=200, w_N=100, the second scanpath is really different and seems to mimic a memorizing scanpath.

Comparison to human data

In order to estimate relevant variable weights w_S, w_D, w_L, w_H, w_N for each process, simulations were run. For each combination of parameters, one artificial scanpath was generated for each participant scanpath. Then each of these pairs of scanpaths were compared. Averaging the result of these comparisons for each combination of parameters gives an overall measure of the adequacy of that version of the model to the human behavior.

Comparison of scanpaths

Comparing scanpaths cannot be done at the level of fixations. Even two humans do not produce identical scanpaths. Higher level comparisons should be performed.

The Levenshtein distance (also called string edit distance) is the most common way of comparing scanpaths (Privitera & Stark, 2000). Each scanpath is encoded as a string of letters in which each letter corresponds to the area of interest (AOI) that each fixation hits. Then the Levenshtein distance between two scanpaths is the number of insertions, deletions or substitutions that are necessary to go from one string to the other. In our case, this method cannot be used as it is: considering each word as an AOI would be inappropriate because it would not consider the spatial relationship between words (on the same line for example).

An interesting method was recently proposed by Jarodzka et al. (2010). Each scanpath is viewed as a sequence of geometric vectors. Each vector corresponds to a saccade in the scanpath. Then a scanpath with n fixations is represented by a set of n-1 vectors. The two sequences that has to be compared are aligned according to their shapes (although the authors note that alignment can be performed on other dimensions): it means that to each vector of scanpath #1 corresponds one or more vectors of scanpath #2, such that the path in the matrix of similarity between vectors going from (1,1) (similarity between first vectors) to (n,m) (similarity between last vectors) is the shortest one. Once the scanpaths are aligned, various measures of similarity between vectors (or sequences of vectors) can be used: average difference in amplitude, average distance between fixations, average difference in duration, etc.

For example, Figure 3a shows the scanpath from participant #13 (first saccade is going upward). The model outputs the scanpath of Figure 3b for a particular combination of variables weights (first fixation is on the first word).

The alignment procedure attempts to match the six vectors (for the six consecutive saccades) of the participant scanpath with the four vectors of the model scanpath. According to Jarodzka's method, the best match is the following: 1-2/1 ; 3/2 ; 4/3 ; 5-6/4 (saccades 1 and 2 of participant scanpath are aligned with saccade 1 of model scanpath, saccade 3 is aligned with saccade 2, etc.).

Once scanpaths are aligned, similarity measures are computed for each alignment. Instead of using Jarodzka's measures of similarity between aligned sequences of saccades which are not fully relevant to the study, the following measures of distance were used:

- the spatial distance between saccades (computed as the distance in pixels between midpoints of each saccades and normalized for the paragraph diagonal). Similar scanpaths should have aligned saccades located in similar regions of the screen.
- the angle between saccades (computed as the normalized cosine between saccades). Similar saccades should have aligned saccades in similar directions.
- the difference of amplitude between saccades (computed as the normalized difference of saccades lengths). Similar scanpaths should have aligned saccades of similar lengths.

On the previous example, the results are: distance between saccades = 0.20; angle between saccades = 0.14; amplitude ratio = 0.38 (AVERAGE = 0.24).

It means that the model with these parameters is quite bad at reproducing the amplitude of saccades. It is however better at reproducing the position of the scanpath and above all the angle between saccades.

With another combination of parameters, another example of artificial scanpath is generated (Figure 3c, the first fixation is on the first word and the third saccade is a regressive saccade).

Alignement with the participant scanpath is now the following: 1/1 ; 2-3/2 ; 4-5/3 ; 6/4-5-6-7-8.

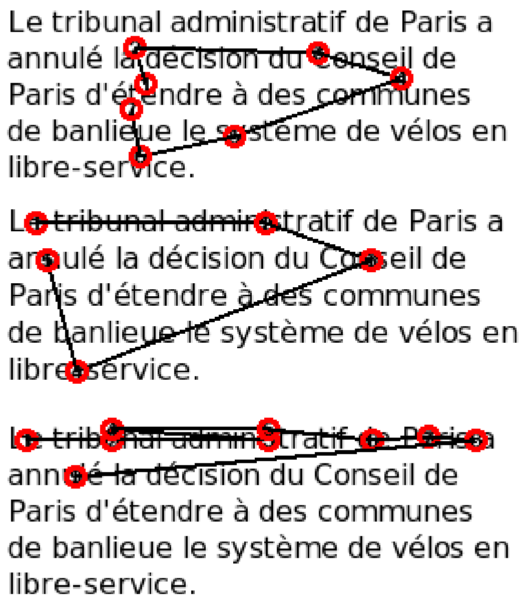

Figure 3: (a) A participant scanpath. (b)(c): Two artificial scanpaths.

Comparison results are: distance between saccades = 0.30; angle between saccades = 0.31; amplitude ratio = 0.39 (AVERAGE = 0.33).

That model is about the same as the previous one predicting saccade amplitudes. However, it is much worse with respect to the shape of the scanpath as well as its position.

Parameter adjustment

In order to estimate the appropriate value of parameters w_S, w_D, w_L, w_H, w_N for the scanning, skimming and memorizing tasks, the average distance between all participant scanpaths and corresponding model scanpaths for each combination of parameters was computed. Actually, only the values of four parameters out of 5 are needed since relative values instead of absolute values are considered. For instance what is relevant is to know that in the scanning condition w_L should be 2 or 3 times higher than w_D. Therefore, one parameter was set to an arbitrary value and the other values that all together produce scanpaths that are similar to human scanpaths were searched. w_N, the memory parameter, that we know cannot have a null value, was set to 100.

In all cases, the first fixation occurred on the first word of the paragraph.

Memorizing condition

In the memorizing condition, there is no target so the w_S parameter is not relevant. After several exploratory simulations on different ranges of values and on a subpart of the data, the following integer values were more carefully tested for the remaining parameters: $w_D \in [0,9]$, $w_L \in [0,5]$, $w_H \in [0,1000]$.

For each of the 1120 participant scanpaths, the generation of the corresponding artificial scanpath was stopped when it reached the same number of fixations.

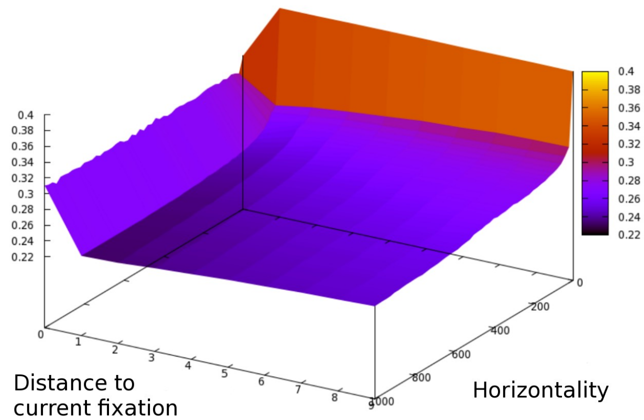

Figure 4: Average distance between human and models scanpaths as a function of distance and horizontality in the memorizing task.

The minimum average distance between model and participants was found for the following values:

w_S=0, w_D=1, w_L=0, w_H=700, w_N=100

It means that moving horizontally is the most important thing. Not going back to previously visited words is important but not that much. The length of words has no effect at all. Making short saccades is not very important.

In order to better understand the effects of variables, Figure 4 presents the average distance between scanpaths as a function of horizontality and distance to the current fixation.

Although minimizing the distance to the current fixation does not seem to play much role, results are much worse when the weight of that variable is set to 0. In the same way, the worst results are found when the horizontality weight is set to 0. Fit to human data increases until about 500 and then becomes about the same.

Scanning condition

The same procedure was performed with the data coming from the scanning task. Parameter w_N was also set to 100. The entire procedure was longer to perform because there is one more dimension to take into account. Each simulation was stopped when the target word was found or when the number of fixations was the same as the number of fixations performed by the participant.

The minimum average distance between model and participants was found for the following values:

w_S=3, w_D=6, w_L=3, w_H=15, w_N=100

The pattern is completely different from the memorizing task. Horizontality is much less important. As expected, similarity of shape plays a role which appears as important as the length of words (although these variables may be dependent from each other since the targets were not short words). Distance to the current fixation also plays a significant role: saccades should not be too long.

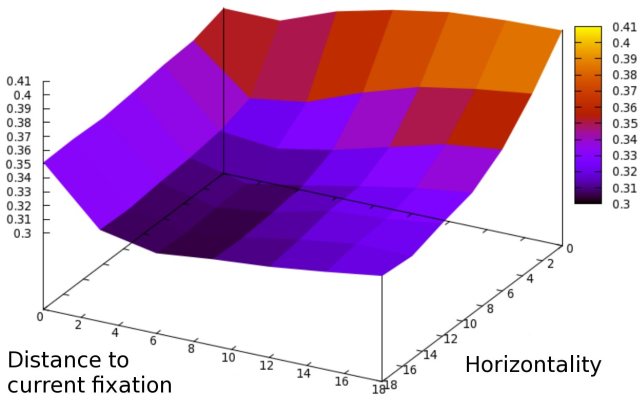

Figure 5: Average distance between human and models scanpaths as a function of distance and horizontality in the scanning task.

Like previously, Figure 5 presents the average distance between scanpaths as a function of horizontality and distance to the current fixation. It shows that results are much worse if the distance weight is set to 0 (meaning that keeping short distance to the current fixation is important) and also if horizontality is close to 0. Even in the scanning task, horizontality of saccades plays a role, but not as much as in the memorizing condition.

Skimming condition

The same procedure was performed with the data coming from the skimming task, whose objective was to make a decision about the semantic relatedness of the paragraph to a goal.

The minimum average distance between model and participants was found for the following values:

w_S=0, w_D=90, w_L=40, w_H=900, w_N=100

Although horizontality of saccades plays a major role, fixating long words is important. This is probably because long words contain more semantic information than short words, which is essential in this semantic task. Distance to the current fixation is even more important. This semantic task requires a comprehension of the text, which often requires reading linearly some group of words, by means of short saccades going from one word to the next one.

Conclusion

This paper presents a model of eye movements on textual material which was applied to 3 different ways of processing a paragraph: searching for a specific word (scanning), assessing the semantic relatedness of that paragraph to a goal (skimming) or memorizing the paragraph. Five parameters were adjusted which showed that:

- length of words plays an important role in the skimming task (40), a reduced role in the scanning task (3) and no role at all in the memorizing task;
- minimizing the distance to the current fixation is crucial in skimming (90), not so important in scanning (6) and slightly necessary in memorizing (1);
- horizontality of saccades is very important in memorizing (700) and skimming (900) but not much in scanning (15);
- visual similarity of word shape is only necessary in scanning (3).

The effects of some variables in different tasks were described and, more important, we provided a model that is able to reproduce the shape of a human scanpath given the task. The next step is to supplement this model with a semantic component. One goal would be to model the way users navigate in a web page (Chanceaux et al., 2009).

The process of searching in the space of parameters was done in a brute force way. Optimization techniques, and especially evolutionary algorithms, to improve that process are under investigation.

Acknowledgments

This work was funded by the French National Research Agency (ANR) under the project GAZE&EEG.

References

Carver, R. (1990). *Reading rate: A review of research and theory*. San Diego, CA: Academic Press Inc.

Chanceaux, M., Guérin-Dugué, A., Lemaire, B., Baccino, T. (2009). A model to simulate Web users' eye movements. In *Proc. of the 12th Conference on Human-Computer Interaction (INTERACT'2009)*, LNCS 5726, 288-300.

Jarodzka, H., Holmqvist, K. & Nyström, M. (2010). A vector-based multidimensional scanpath similarity measure. *Proceedings of the 2010 Symposium on Eye Tracking Research & Applications* (pp. 211-218), ACM.

Landauer, T., McNamara, D., Dennis, S., & Kintsch, W., (2007) *Handbook of Latent Semantic Analysis*. Lawrence Erlbaum Associates.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions and reversals, *Soviet Physics-Doklady* 10, 707-710.

Martin, D. I. & Berry, M. W. (2007). Mathematical foundations behind Latent Semantic Analysis. In Landauer et al. (Eds.) *Handbook of Latent Semantic Analysis*. Lawrence Erlbaum Associates.

Privitera, C. M. & Stark, L. W. (2000). Algorithms for defining visual Regions-of-Interest: Comparison with eye fixations. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(9), 970-982.

Reichle, E. D., Rayner, K. and Pollatsek, A. (2003). The E-Z Reader model of eye-movement control in reading: Comparisons to other models. *Behavioral and Brain Sciences*, 26, 445-526.

Simola, J., Salojärvi, J. & Kojo, I. (2008). Using hidden Markov model to uncover processing states from eye movements in information search tasks. *Cognitive Systems Research*, 9, 237-251.